

## **REMARKS**

Claims 1-42 are pending in the application. Claim 42 is added. Claims 1, 26, and 27 are amended. Claim 40 is also amended. Paragraph [0032] of the specification is further amended to correct a typographical error. Support for the amendments and additions can be found in the originally filed specification. The Examiner is respectfully requested to reconsider and withdraw the rejection(s) in view of the amendments and remarks contained herein.

### **REJECTION UNDER 35 U.S.C. § 103**

Claims 1, 2, 8-29, and, presumably, 40 and 41 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Ewert et al. (U.S. Pat. App. Pub. No. 2001/0034586) in view of Bakke (U.S. Pat. No. 6,330,621). Claims 3-7 and 30-39 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Ewert et al. (U.S. Pat. App. Pub. No. 2001/0034586) in view of Bakke (U.S. Pat. No. 6,330,621) and Nozaki (U.S. Pat. No. 6,128,644). These rejections are respectfully traversed.

For reasons further discussed below, Ewert et al., Bakke, and Nozaki do not teach, suggest, or motivate “a controller that: (1) receives at least one of the operation requests, (2) finds available and capable devices appropriate for an operation specified by a received operation request and forms a set of candidates therefrom, wherein each candidate contains one or more of the founds devices based on (i) the received operation request, (ii) a description of a media flow specified in the operation request, and (iii) load characteristics of the one or more devices, (3) assigns a score to the one or more devices by using: (i) information from load characteristics of the one or more devices and networks in a networked surveillance system, (ii) descriptions of current

media flows stored by said flow information service, and dynamically configurable user preferences, and (4) allocates the one or more devices for the requested operation by selecting the one or more devices from a list of the candidates ranked according to the scores” as recited in amended claim 1. Claims 26 and 27, especially as amended, recite similar subject matter. Therefore, Ewert et al. Bakke, and Nozaki do not teach, suggest, or motivate all of the limitations of the independent claims. These differences are significant for numerous reasons. Example reasons are that the Examiner’s suggested modification of the teachings of Ewert et al. by the teachings of Bakke and Nozaki would result in: (1) determination of a score value of for a candidate device that is only determined by a value returned from an NCost function, and (2) allocation of a busiest data storage device. Further it is not possible to combine these three references to establish one (record) of many surveillance operations. These and other points are discussed by the inventors in detail below.

Accordingly, Applicants respectfully request the Examiner reconsider and withdraw the rejections of independent claims 1, 26, and 27 under 35 U.S.C. § 103(a), along with rejection on these grounds of all claims dependent therefrom. Applicants note that new claim 42 depends from claim 1 and should be allowed for the same reasons.

#### **Detailed Discussion**

We have studied Ewert, Bakke, and Nozaki and try to modify Ewert in view of Bakke and Nozaki as suggested in objections to realize the functionality of disclosed application based on our understanding of their teachings.

In the following we try to include intelligent data manager into controller of Ewert for allocation or creation of logical data storage device (MM Recorder) for write (recording) of user data object (media flow) for MM Recorder's record operation (one of the surveillance operations). We will invoke Nozaki when needed.

Bakke teaches allocation logical/virtual data storage device (here on we use logical in meaning of virtual and logical) or creation of new logical data storage device by combination of logical storage device based on policy attributes of user/application data object (file) in which policy attributes of user data object (we assume "user" and "application" is the same hereon) are related to storage cost(Cost per MB(lg)) and access pattern (Time to First byte(lg), I/O read/write speed, etc.). Bakke teaches that each logical data storage device has fixed policy attributes values (see, e.g., col.5, lines 40-60). Bakke teaches the usage of distance calculation between the policy attributes of user data object and the policy attributes of logical data storage device for allocation or creation of logical data storage device.

Bake does **not teach** how the "weight value" to attributes is used. We assume that the distance calculation in (see, e.g., col. 6, lines 25-26) in view of the discussion in (see, e.g., col. 4, lines 15-35) may be mean that the following distance calculation

$$\sqrt{w_x * (x1 - x2)^2 + w_y * (y1 - y2)^2 + w_z * (z1 - z2)^2}$$

Where: x1 is the attribute value defined by user policy of user data object, x2: is the attribute value defined for the logical data storage device in fixed policy attributes of device, and  $w_x, w_y, w_z$  are the weight associated with each attribute based on our understanding/assumption. Furthermore, we remove the "square root" in view of (see, e.g., col. 6, lines 40-45), and the final distance calculation of Bakke becomes;

$$\left[ w_x * (x1 - x2)^2 + w_y * (y1 - y2)^2 + w_z * (z1 - z2)^2 \right]$$

The dimensions (attributes) in policy of logical data storage device should match with the (attributes) in policy of user data request to make sure the distance calculation returns meaningful value.

At this point, this distance calculation formula is different from what recited in Claim-40;

$$\begin{aligned} S_i = & W_{nc} * NCost(src, i) \\ & + W_{dpl} * DataPlacementRule(src, i) \\ & + W_{load} * (W_{bw} * BW_i + W_{ds} * DS_i + W_{conc} * Conc_i); \end{aligned}$$

The given formula in Claim 40 and explained in [0031] does not calculate the distance (in teachings of Bakke) between policy attributes of logical data storage device and policy attributes of user data object. Can it be modify to accomplish the operation?

We try to use Bakke's distance calculation to find the best fit MM Recorder (logical data storage device) to record a multimedia flow (user data object). Furthermore, we only consider the allocation function not the creation function since Bakke does not teach a method that creates a new logical data storage device from existing logical data storage devices (Bakke only gives examples, such as RAIT by parallel combination of 4 TAPES).

To be able to process recording of media flow into MM Recorder, we assume that media flow is user data object (examiner's point) and MM Recorder is logical data storage device (examiner's point), and system is trying to allocate the best fit MM

Recorder with the above revised distance calculation. In this example, Media flow comes from camera and it has certain bandwidth requirement (MB/sec)(such as, MPEG2, MPEG4 media streams [0032]).

We could not find such policy attribute in Bakke but we included as an additional attribute into policy of user data object (media flow) as a user requirement according to teachings of Bakke. Then, we realize that the maximum network bandwidth (NIC card) of logical data storage device (MM Recorder) is needed so that we can compare these dimensions in distance calculation of Bakke. Then, we add maximum network bandwidth as a new attribute into policy set of logical data storage device (MM Recorder). It is ok since this value is fixed according to teachings of Bakke. Then we tried to use the distance calculation formula to allocate logical data storage device.

For example, MM Recorder-1 with maximum network bandwidth capacity of 10, MM Recorder-2 with maximum network capacity of 8, and media flow with 2. Although, we do not know whether “bandwidth requirement” of user data object (media flow) should be compared against the “maximum network bandwidth” of logical data storage device (MM Recorder), we proceed with the distance calculation assuming they represent the same attribute of a policy. Based on distance calculation and set all the other dimensions (attributes) of logical data storage devices (MM Recorders) to the same value (and maybe use weight=0 for not used dimensions), MM-Recorder-1 received  $(10-2)^2=64$ , and MM-Recorder-2 received  $(8-2)^2=36$ . According to Bakke’s teaching, system should select MM-Recorder-2. At this point we do not know whether it is a good or bad choice but it might work if each recorder handles only one write operation at a time (no concurrency).

MM Recorders in surveillance system handles multiple read/write operations (record/playback operations) concurrently. To realize this aspect, we added current bandwidth usage information (as indication of concurrently running operations) into policy set of logical data storage device (MM Recorder). Since the value of this attribute cannot be fixed, we disregard the Bakke's teaching that the values of attributes in policy of logical data storage device (MM Recorder) are fixed (due to their physical nature). Then, we need a way to collect and update this information in policy manager in teachings of Bakke.

To collect such data, we invoke Nozaki, "server status notification means" to collect such data and assumed that policy manager of intelligent data storage manager of Bakke and server status management means of Nozaki are combined and implemented by "controller" of Ewert. We tried to use 'LOAD' definition of Nozaki (number of (TCP) IP packets sent to WWW Server per unit time) for **identical WWW Servers** (requirement of load distribution in Nozaki). We have quite a conflict at this point (when we put Bakke and Nozaki together), since, we do not know "how requirement of load manager that WWW Servers should be identical" can be combined with the intelligent data storage manager (of Bakke) where logical data storage devices are allocated or created from combinations of physical/logical data storage devices where clearly at least TAPE, DISK, and OPTIC, are not identical (as well as the created new logical data storage device from combination of logical data storage devices). Bakke teaches that data storage devices have different data characteristics. Also, there is an issue on how to create logical WWW Server (data storage device) from

combination of WWW Servers (logical data storage devices) to match with the requirements of user data object (HTTP Request, or, file on WWW Server?).

We assumed that these conflicts could be overlooked. Then, MM Recorder (WWW Server) receives a record request (we assume this request is in TCP/IP packet since Nozaki is using HTTP protocol, WWW Client/Browser and WWW Server), starts receiving user data object (media flow, for example multicasted media data from camera) from network, and writes media flow data into data storage (this one is real/physical data storage). We realized that we need to collect current bandwidth usage on WWW Server (MM Recorder) and we cannot collect this by simply counting number of IP packets sent to MM Recorder (WWW Server).

We have changed the 'LOAD' definition of Nozaki as number of IP packets **sent to** and **received by** WWW Server per unit time. We realized that it cannot account for multicast traffic since it will not have the destination IP Address at all. Only the WWW Server would know such information.

Although Nozaki's teachings on balancing load without changing WWW Server and WWW Browser (that is main reason of introduction of 'local proxy server' running on the machine in which WWW Client is running), we put the "server status notification means" into all the machines running WWW Server and revise the definition of LOAD in Nozaki and method of obtaining this information on WWW Server by counting number of in/out packets per unit time on the machine in which WWW Server is running. [We also assume that we only have one WWW Client (controller) since we could not figure out how do we realistically send these notifications all possible WWW Clients in internet

assuming even registration means is used to subscribe such notifications since there are lots of WWW Clients and Servers out there]

We invoke teachings of Ewert “unsolicited status messages from I/O hardware to controller”, assumed that “I/O hardware” could be a logical data storage device or WWW Server (MM Recorder), assumed that this status message (as examiner pointed it out the detail content of “unsolicited status message” is not specified in teachings of Ewert) could contain such information in view of Nozaki with the revised LOAD definition and method to obtain, and the value of attribute in policy manager could be updated by controller of Ewert (assuming) periodically. Then, we try to see how this system allocate or create a logical data storage device (MM Recorder) to write (record) user data object (media flow).

For example, MM Recorder-1 has maximum network bandwidth capacity of 10 and current bandwidth usage of 7.1, MM Recorder-2 has maximum network capacity of 8 and current bandwidth usage of 7, and media flow with 2 (user policy). We tried to use maximum bandwidth attribute in distance calculation (assuming bandwidth requirement of media flow can be compared) and MM-Recorder-1 received  $64=(10-2)^2$  and MM-Recorder-2 received  $36=(8-2)^2$ . Clearly, MM-Recorder-2 cannot handle the media flow since MM-Recorder-2 does not have enough available bandwidth ( $7+2>8$ ).

We tried to use current bandwidth usage in comparison with bandwidth requirement of user data object (media flow) to overcome this problem, then, MM-Recorder-1 received  $(7.1-2)^2=26.01$  and MM-Recorder-2 received  $(7-2)^2=25.00$  assuming all the other dimensions has the same value. The distance calculation picks MM-Recorder-2 again ( $7+2>8$ ).



Furthermore, we realized that the distance calculation may not be able to pick the best one (Bakke does not teach how to pick the best one). We changed the current bandwidth usage of MM-Recorder-1 to 7 (instead of 7.1), then the results of distance calculation for both logical data storage devices (MM Recorder) become 25 (regardless of whether the logical data storage device can handle write operation or not at the time of request). Now which one is the best? Then, we realized that all available logical data storage devices (listed in intelligent data storage manager) have an associated distance to the given policy of user data object in n-dimensions. In FIG.2 of Bakke's example, the distance calculation will generate a distance value to all logical data storage devices (TAPE, DISK, and OPTICAL). Furthermore, as discussed above there could be multiple such devices (MM-Recorder-1 and MM-Recorder-2 that both received the same distance value) with the same distance value, and each one is as good as another one based on distance calculation (yet, MM-Recorder-2 is wrong choice).

Further study shows that (whether weighted or not), for a given point (user request in FIG.2 of Bakke) in n-dimensional space (each dimension is an attribute in policy of logical data storage device and user policy/requirement as taught in Bakke) and distance value of  $d$ , there could be many points having the same distance value to the given point (user requirement of user data object). When  $n=2$ , these points are on a circle. When  $n=3$ , these points are on a sphere. All such solutions (points) are as good as another (point on the circle or sphere) since all have equal distance to the user request (given point). The selected logical data storage device out of logical data storage devices with equal distance to user requirement is arbitrary tradeoff between (for instance) "Cost of MB" and "Time to First Read" (or any other different number of

combinations of attributes in policy of logical data storage device and user data object with or without weighting). We are a little bit confused about teachings of Bakke related to “the intelligent storage manager must resolve between conflicting priority levels,”(see, e.g., col.4, lines 29-32) and “the intelligent data storage manager uses weighted values that are assigned to each of presently defined logical devices to produce a best fit solution to the requested policies in an n-dimensional best fit matching algorithm.” (Abstract).

Then, we return back to teachings of Bakke about the “matching” step and how the system decide to create a new logical device considering that the “matching” step may indicate that MM-Recorder-2 does not have enough network bandwidth capacity (at this moment) to write (record) user data object (media flow) so that the above anomaly does not happen. We looked at the example in teachings of Bakke (see, e.g., col.4, lines 52-65) related to FIG.2 to understand how Bakke decides that “the user request” does not match the data storage characteristics of any of the devices. In (col.4 lines 52-65) or (col. 6, lines 60-68 related to example in TABLE-2), Bakke does not teach a method that decides there is no match between policy of logical data storage device and policy of user data object. If Bakke is using the weighted values of “distance calculation” as discussed before, this distance always exists. By using given “distance calculation”, such decision could only be made by introduction of “maximum allowed distance” as threshold value to see if there exist any point (policy of presently available logical data storage devices) within (less than or equal to) this distance threshold value. Bakke does not teach this either. (Is it obvious how to set such a distance threshold value?).

We decided to introduce candidate selection step (before the distance calculation) to eliminate the logical data storage devices who cannot able to fulfill the request by comparing their available resource attributes with respect to the attributes (requirements) of user data object (media flow). At this point, these attributes logically do not match but we introduced “dimension” as another abstract entity and mapping from an attribute of user request (such as media bandwidth requirement) to a dimension and mapping from an attribute of device characteristics (such as available network bandwidth) to the same dimension so that we could have meaningful measures (for distance calculation). Now we can eliminate MM-Recorder-2 (due to candidate selection step we have just introduced, not the weighted distance calculation) and safely pick MM-Recorder-1 for the record operation.

Now, we were trying to capture “concurrency” attribute of logical data storage device (MM Recorder). The number of maximum concurrent operations on device is restricted due to various physical limitations, such as number of I/O devices (see, e.g., col. 5, lines 1-11, RAIT example in Bakke), number of threads, etc. Also, from the discussion before, we will need number of currently running operations similar to current network bandwidth usage. Bakke does not teach this attribute in the policy of logical data storage device (and it cannot be fixed value contrary to Bakke’s teachings for values of policy attributes of logical data storage device), then we invoke teachings of Nozaki to measure number of concurrent operations, add additional attributes/columns (maximum number of concurrency and number of currently running operations) of policy of logical data storage device (WWW Server, MM Recorder) in intelligent data storage

manager (of Bakke) and server status manager means (of Nozaki) implemented by controller of Ewert.

We study the definition of LOAD in Nozaki whether to see LOAD measurement of server status notification means can measure the number of concurrently running operations (HTTP Requests) in WWW Server (MM Recorder). LOAD that is “The number of IP packets sent to WWW Server per unit time” in teachings of Nozaki cannot measure the number of concurrently running operations on WWW Server. It can only measure the number of HTTP Requests (IP Packets) sent to WWW Server per unit time provided that each (TCP in HTTP) IP packet contains only one HTTP request (this could be HTTP GET) and the reply of request always fits into only one IP packet in response to the request. If the request is HTTP POST request, depending on the amount of data being POSTed, the HTTP POST request could require more than 1 IP packets. Thus, the number IP packets sent to WWW Server has nothing to do with the number of HTTP Requests being processed by WWW Server (in case of HTTP POST). If HTTP GET request is used (counted) provided that HTTP GET request message always fits into one IP packet, the response of GET request may or may not fit into a single IP Packet depending on the volume of user data object (file) on WWW Server (logical data storage device). When HTTP GET requests for large files (user data objects), response will introduce a lot of packets outgoing from WWW Server (and it also has impact on CPU load/usage). Even for short files (user data object) assuming it can fit into one IP packet, the behavior/response time differs whether the user data object is cached in WWW Server or user data object requires disk access. Then, we looked at whether we can use the selection method of MM Recorder (WWW Server) by using the smallest relative

amount of processing, is obtained by dividing the “load” value by the “ratio” value as taught by Nozaki. According to teachings of Nozaki, the value of “RATIO” represents the **processing rates relative to the slowest WWW Server in WWW Servers**, in which “RATIO” of the slowest WWW Server 302 having reference value of “1” (see, e.g., col.9 lines 61-68 continues col. 10, 1-18). We have used their host names in the following example;

	LOAD	RATIO	Relative Amount of Processing (L/R)
Host1.a.co.jp	9	3	3
Host2.a.co.jp	2	1	2

The “server status notification means” collect number of IP packets sent WWW Server per unit time. Assume that Host1.a.co.jp is 3 times faster than the slowest WWW Server (Host2.a.co.jp) (RATIO is configured by system administrator properly, see, e.g., col. 10, lines 64-67 continues col. 11, lines 1-20). According to teachings of Nozaki, we assigned relative value of “1” to the slowest WWW Server (Host2.a.co.jp).

Assume that between measurement intervals (per unit time), Host1.a.co.jp received 9 requests for retrieval (HTTP GET) of short user data object(file), each request fits into one IP packet and response fits into one IP packet and all of them are completed within measurement interval since requested files are short and possibly

cached and Host1.a.co.jp is fast. Host2.a.co.jp received 2 requests for retrieval (HTTP GET) of large volume of user data objects (file), and, Host2.a.co.jp is still trying to send the IP packets of large volume of user data object (file) to requestor since the beginning of the measurement interval (since these measurements are done per unit time) and these two operations are still running.

When a new HTTP GET request (record to MM Recorder) comes at the beginning of the next measurement interval (while Host2.a.co.jp is still serving 2 HTTP GET requests), can the request distribution section determine a server (WWW Server, or, logical data storage device, or, MM Recorder) experiencing less load, based on the load information (Abstract of Nozaki)? NO. The method taught in Nozaki to select MM Recorder (WWW Server) that has the smallest relative amount of processing (in Nozaki) obtained by dividing the “load” value by the “ratio” value (taught by Nozaki) selects MM Recorder-Host2.a.co.jp over Host1.a.co.jp (even Host1.a.co.jp machine is totally idle and 3 times faster than Host2.a.co.jp). The usage of L/R is not efficient.

We assumed that this is not a problem and try to see how we can measure. Finally, we concluded that the concurrency measurement can only be done on machine running WWW Server. We put server status notification means into all machines running **identical** (requirement of load distribution principle in teachings of Nozaki) WWW Server in WWW Servers. The measure of ‘number of packets in/out on machine’ does not capture the number of concurrently running operations in WWW Server since the number of IP packets into WWW Server (assuming one packet is one request) does not able to measure how many GET operations are actively in service by the WWW Server. Then, we put the server status notification means as a function of WWW Server

(although, this is contrary to Nozaki's teachings since no change in WWW Server and WWW Client) since WWW Server process should know the number of concurrently running operations.

Then, we try to use concurrency measurements in distance calculation taught by Bakke. There is nothing we can put into policy set of user data object as a meaningful attribute that could be compared against the policy set of logical data storage devices.

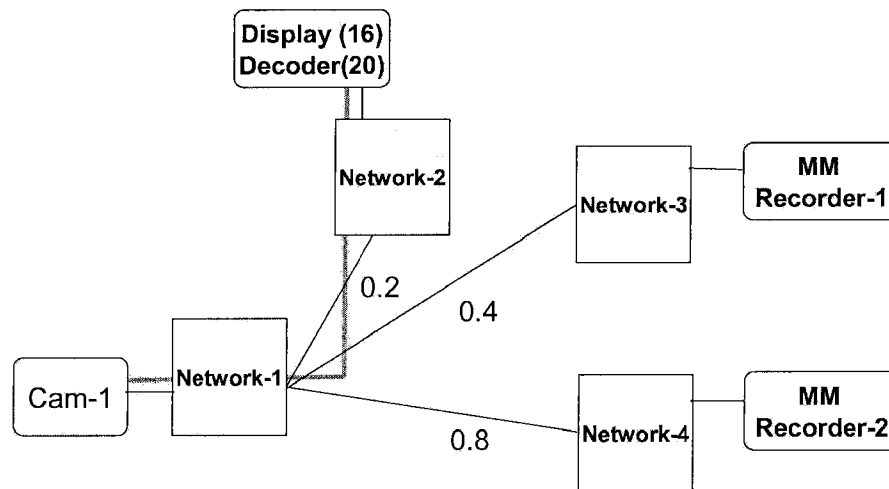
At this point, we turn back to the formula recited in Claim 40;

$$\begin{aligned} S_i = & W_{nc} * NCost(src, i) \\ & + W_{dpl} * DataPlacementRule(src, i) \\ & + W_{load} * (W_{bw} * BW_i + W_{ds} * DS_i + W_{conc} * Conc_i); \end{aligned}$$

The load on the candidate device is calculated by weighted sum of parameters, such as  $BW_i$ . And they are calculated by taking ratio of currently available and the maximum capacity. This definition of "RATIO" is not taught by Nozaki since RATIO assigned to MM Recorder (WWW Server) by system administrator is relative with respect to the reference value of 1 that is assigned to the slowest MM Recorder (WWW Server) in MM Recorders (WWW Servers with identical functions) . Logarithmic approach applied to some policy attributes, such as Cost per MB, Time to first byte, described by Bakke is not used either.

#### Flow Information, and Network Proximity Matrix, NCost example

The following example depicts the location of devices on the networked surveillance system. Here, we will look at function of NCost parameter in Claim-40.



In this setting, the “red” arrow represents the multicast media flow to the Display Device (such as plasma TV) through Decoder device. This media flow information is available in Flow Information service (in Claim-1). When a record request (Cam-1) comes to resource manager of claim-1, it needs to pick one MM Recorder.

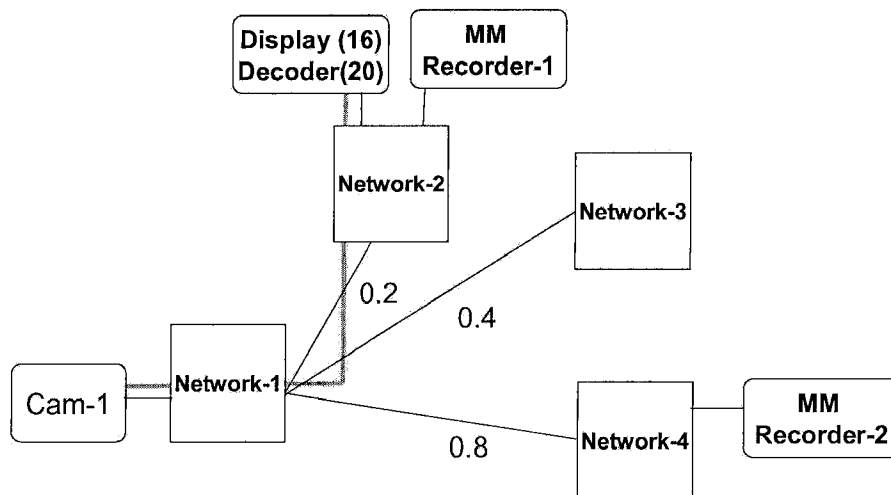
In the above problem, we also assume MM Recorder-1 (attached to Network-3) and MM Recorder-2 (attached to Network-4) has the exact values for other parameters (load and data placement rules) of the score function in Claim-40 except return value of NCost by using the information in Network Proximity Matrix.

Without considering whether media flow of camera in a given network, NCost assigns 0.4 as a value between Network-1 and Network-3, 0.8 as a value between Network-1 and Network-4, and 0.2 as a value between Network-1 and Network-2. The interpretation is that candidate devices attached to Network-4 is preferred over candidate devices attached to Network-3 or Network-2, and candidate devices attached to Network-3 is preferred over candidate devices attached to Network-2.



In this setting, the allocation of MM Recorder is depends on NCost since resource manager looks to Flow information service and finds out that the Cam-1 media flow only goes to Network-2 while available MM Recorders are on Network-3 and Network-4. The selection of MM Recorder in this case is based on value from NCost() function. As a result, the resource manager prefers MM Recorder-2( $S_2=0.8$ ) over MM Recorder-1( $S_1=0.4$ ). Assuming that all other parameters in formula returns the same value, the score value of candidate device is only determined by the value returned from NCost function.

In the following, we have moved MM Recorder-1 to Network-2.



Since the media flow of Cam-1 is already in Network-2, NCost function returns 1 for candidate devices in Network-2 and resource manager prefers MM Recorder-1 ( $S_1=1.0$ ) over MM Recorder-2 ( $S_2=0.8$ ). Assuming that all other parameters in formula returns the same value, the score value of candidate device is only determined by the value returned from NCost function.

The Bakke distance calculation picks the busiest data storage device;

	Max NW I/O BW (C1)	Current NW I/O BW(C2)	Avail NW I/O (C3)	Request NW I/O (C4)	D(C1,C4)	D(C1,C4)	Avail/Max
MM1	10	6	4	2	4	64	<b>0.4</b>
MM2	8	6	2	2	<b>0</b>	<b>36</b>	0.25

With the  $BW = \text{Avail}/\text{Max}$  definition,  $MM1.BW = 4/10 = 0.4$  and  $MM2.BW = 2/8 = 0.25$  and MM1 is selected.

### Conclusion

We believe that we have established the difficulties to combine cited 3 references to establish one (record) of many surveillance operations in disclosed application.

### Comments on Ewert, Bakke, Nozaki

Nozaki

1. no concept of each server's actual experienced load due to the operations on devices/components of surveillance system. For example, multicasted media flow (for example, variable rate mpeg stream(such as MPEG2, MPEG4)) can be captured the method of obtaining load of device
2. (load !=BW in Claim-1,40) the method of load measurement (related to number of packets send to WWW Server) is not indicative of network bandwidth usage of

device itself. First, the amount of output of device is important. Second, an IP packet sent to server does not mean that server accepted and processed the packet. This measurement can only be done by device itself not by another program in IP segment.

3. (load!=number of concurrently running operations) The method of load measurement cannot capture this at all. For example, the serving of HTTP GET operation in WWW server is more related to the amount/volume of HTML page and/or data/file send from WWW server to the requesting WWW Client. If the file is short (then, it fits into one IP frame) and/or cached by WWW server, the LOAD column will be quite high (although it does not count the number of IP packets from WWW Server but this example to make 1 In-1 Out). If the file is too long, WWW server needs to send a lot of IP packets (carrying the file data) and the duration of GET operation will be different depending on the file itself. When WWW Clients ask such big files, the method of measuring LOAD does not able to capture the number of concurrent running GET operations nor the actual load experienced by the WWW server (MM Recorders, Analyzers, etc.) to accomplish these (surveillance) operations. (Hasan: It is also possible to show that LOAD/RATIO, does NOT necessarily pick the lowest loaded server).
4. (RATIO!=BW or other measures in Claim-40) RATIO is against the slowest WWW server in WWW Servers in which the slowest WWW Server ratio having reference value of 1. The calculation of BW, etc. attributes of load in Claim-40 is with respect to the device itself and ratio between available capacity of these parameters and their respective maximum capacities in device itself. The usage

of RATIO is NOT the relative to MM Recorder in MM Recorders with respect to the slowest MM Recorder.

5. The selection criteria of LOAD/RATIO has nothing to do with the selection criteria in Claim-40,41. It does not capture the definition in Claim-40 and It CANNOT capture all the aspects (even not a single one) specified in Claim-40,41.
6. No concept of “Flow Information” and/or Network Proximity Matrix. The location of WWW Client has no importance for selection of WWW Server in Nozaki. It is obvious that HTML pages are not streamed in internet. These files/user(application) data objects (in Bakke) stored in (identical) WWW Servers (in Nozaki) to be fetched by WWW Clients.

#### Bakke

- 1- It is static allocation of storage space (stored by single or multiple storage devices such as Tape, Disk, RAID, RAIT, etc). No notion of media flow. Is the media flow the same as user data file to be stored on virtual/logical storage devices? The similarity of writing/storing a file operation to MM Recorder does not show the media flow (such as camera sending image data, such as streamed mpeg with transport protocol) at all. The operation of Analyzer is different than the MM Recorder.
- 2- Does not teach (nor concerned) about the “concurrency” of operations on MM Recorder. The MB/sec to DISK, Tape, RAIT, RAID does not indicates how many parallel operations could be handled by storage devices. As indicated before, it is static allocation of storage space (stored by single or multiple storage devices

such as Tape, Disk, RAID, RAIT, etc.). The media flow is active video stream in the networked surveillance system.

- 3- The usage of Euclidean distance to match does not enable to pick the MM Recorder for storage of data. Assume required BW of media flow (user data object) is 2, there are two MM Recorders; one has available BW 1 and another has available BW 3. Their  $(x1-y1)^2$  definition produces the same value for BW dimension (assuming all the other attributes/dimensions of policy are the same). But, said first MM Recorder is not able to handle the media flow for recording operation at all. (nor it can capture what is said in “flow information” service).
- 4- The LOAD/RATIO from Nozaki cannot be used with Bakke to realize what has been claimed either. Inclusion of LOAD, RATIO as policy attributes of data storage device still does not come close to what has been claimed in Claim-40.
- 5- Usage of status (in Ewert) with LOAD/RATIO (in Nozaki) to match (in Bakke) with L2 formula does not accomplish what has been claimed in Claim-40, 41. No one accounts for “Flow Information” or “Network Proximity Cost Matrix” definitions (even though the description/method of terms fails to match with the descriptions in the application).

Ewert discloses a system for monitoring and controlling home security and other physical plant functions of households and specifically to support remote access to the system functions via network. Disclosed system may also provide notification of any problems perceived by the system by e-mail message, via telephone page, or audible indicator on or off the premises. Ewert discloses the following functions;

- 1) user controls devices in households via network (through HTTPD),

- 2) user views recorded or live images from monitoring devices via network (through HTTPD, IFXD, GRABD),
- 3) user views status of devices via network (through HTTPD),
- 4) system records the data (including video, audio) obtained from devices (IFXD),
- 5) system records the data (including video, audio) obtained from devices (IFXD),
- 6) system returns list of image file names in response search request operation, containing criterions times, reasons, channels, and the like(page 3, paragraph [0034]),
- 7) returns image in response to search request operation, containing criterions times, reasons, channels, and the like(page 3, paragraph [0034])

Bakke discloses the intelligent data storage manager functions to **combine** the **non-homogeneous physical devices** contained in a **data storage subsystem** to **create a logical device** with **new and unique quality of service characteristics** that **satisfy** the criteria for the **policies** appropriate for the **present data object**. If there is presently no logical device that is appropriate for use in storing the present data object, the intelligent data storage manager **defines a new logical device using** existing **physical and/or logical device definitions** as component building blocks to provide the appropriate characteristics to satisfy the policy requirements. The intelligent data storage manager **uses weighted values** that are **assigned to** each of the presently **defined logical devices** to **produce a best fit** solution to the **requested policies** in an n-dimensional best fit matching algorithm. The resulting logical device definition is then implemented by **dynamically interconnecting the logical devices** that were used as the components of the newly defined logical device **to store the data object**.

Teachings of Bakke aim to solve the storing increasing volume of application data into data storage subsystems in view of the rapid changes in data storage characteristics of the data storage elements and the specific need of the applications' data. Bakke teaches a way to create a logical (storage) device with new and unique quality of service characteristics that satisfy the criteria for the policies appropriate for the present data object (user/application's data object) by combination the non-homogeneous physical (storage) devices and/or logical (storage) devices since the logical device provides a transparency (col. 3, lines 22-25) for a physical storage device (such as tape, disk, etc.) and composite storage device. Bakke teaches the matching between policies of application data and available (logical/physical) storage devices by using euclidian distance (col.6 lines 25-60) between fixed attributes of logical storage device(col.5, lines 42-44) and policy attributes of application/user data object (col. 3, lines 55-58 ). The attributes of logical data storage device contains speed of access to first byte, level of reliability, cost of storage, probability of recall, and expected data transfer rate (col. 3, lines 40-45). Bakke teaches that a best fit logical storage device is selected from rank ordered distance calculations, between 1) the set of attributes for each (logical storage) device and 2) the set of attributes for a file (defined through user policy) (col.6, lines 20-25). Bakke mentions that the algorithm uses "weighted values" (col. 3, lines 30-35, Abstract).

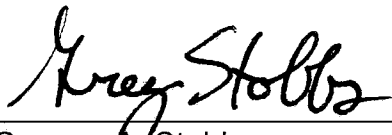
## **CONCLUSION**

It is believed that all of the stated grounds of rejection have been properly traversed, accommodated, or rendered moot. Applicant therefore respectfully requests that the Examiner reconsider and withdraw all presently outstanding rejections. It is

believed that a full and complete response has been made to the outstanding Office Action and the present application is in condition for allowance. Thus, prompt and favorable consideration of this amendment is respectfully requested. If the Examiner believes that personal communication will expedite prosecution of this application, the Examiner is invited to telephone the undersigned at (248) 641-1600.

Respectfully submitted,

Dated: Dec 5, 2007

By:   
Gregory A. Stobbs  
Reg. No. 28,764

HARNES, DICKEY & PIERCE, P.L.C.  
P.O. Box 828  
Bloomfield Hills, Michigan 48303  
(248) 641-1600

[GAS/JSB/kup]